CLAIMS

1. A method for performing failure recovery in a Java platform, comprising the operations of:

executing an application having a service module and a control module, wherein the control module includes application-specific polices for the application;

reporting a detected error in a system component to a runtime executive;

isolating the system component; and

notifying the control module of the system component failure.

2. A method as recited in claim 1, wherein the system component is the service module.

3. A method as recited in claim 2, wherein isolating the system component includes disabling the service module.

4. A method as recited in claim 1, wherein the system component is a Java server.

5. A method as recited in claim 4, wherein isolating the system component includes stopping execution of the Java server.

6. A method as recited in claim 1, wherein the runtime executive is used to facilitate isolating the system component.

7. A method performing failure recovery in a Java platform, comprising the operations of:

executing an application having a service module and a control module on a first Java server, wherein the control module includes application-specific polices for the application;

detecting an error in a system component;

determining which modules are affected by the detected error using the control module; and

restarting the modules affected by the detected error on a second Java server using the control module.

8. A method as recited in claim 7, further comprising the operation of allocating a new module for a module affected by the detected error on the second Java server.

9. A method as recited in claim 8, further comprising the operation of loading a state for the new module from a repository.

10. A method as recited in claim 9, wherein the state is a state of the module affected by the detected error.

11. A method as recited in claim 10, further comprising the operation of enabling routing of incoming messages to the new module.

12. A method as recited in claim 11, further comprising the operation of disabling routing of incoming messages to the module affected by the error.

13. A method as recited in claim 12, further comprising the operation of stopping execution of the module affected by the error.

14. A system for performing failure recovery in a Java platform, comprising:

an application having a service module and a control module, wherein the control module includes application-specific polices for the application;

an error correlator capable of determining which system component includes a reported error; and

a runtime executive in communication with the application and the error correlator, wherein the error correlator reports the system component having the reported error to the runtime executive, and wherein the runtime executive isolates the system component.

15. A system as recited in claim 14, wherein the runtime executive notifies the control module of the system component failure.

16. A system as recited in claim 15, wherein the control module determines which modules are affected by the detected error.

17. A system as recited in claim 16, wherein the control module restarts the modules affected by the detected error.

18. A system as recited in claim 17, wherein a new module is created for each module affected by the detected error.

19. A system as recited in claim 18, wherein the new modules are created on a single new Java server.

20. A system as recited in claim 18, wherein the new modules are created on multiple new Java servers.

21. A system as recited in claim 18, further comprising a repository, wherein a state for each of the new modules is loaded from the repository.

22. A system as recited in claim 21, wherein each state is a state of a module affected by the detected error prior to a time when the detected error occurred.